REMARKS

In view of the foregoing amendments and the following remarks, reconsideration and allowance of this application is requested. Claims 1-22 remain pending, with claims 1, 15, and 21 being independent.

The specification and drawings have been amended in response to the Examiner's objections. In the drawings, the Examiner objected to the use of reference character 308 in Figure 3 being used to designate both 308 and 310. Figure 3 has been amended to replace the second reference character 308 with reference character 310. In the specification, the Examiner objected to the use of "routine 400" on page 7, line 7 because this description does not match the corresponding labeling of Figure 4. The specification has been amended to make it clear that "routine" corresponds to item number 402 in Figure 4. No new matter has been introduced.

Claim 5 has been amended in response to the Examiner's rejection under 35 USC 112.

Claims 1, 15, and 21 recite receiving an identifier and receiving output from a command line utility. The command line utility output is stored in system storage at a location identified by the identifier.

Independent claims 1, 15, and 21 stand rejected under 35 USC 102 as anticipated by Buxton. Applicant requests reconsideration and withdrawal of these rejections because Buxton does not describe or suggest receiving output from a command line utility. Buxton also fails to describe or suggest storing the command line utility output in a system storage at a location identified by the identifier.

Buxton describes a system that allows users of small software applications such as spreadsheets and file viewers to customize these applications to perform the users own preferred tasks. Buxton describes customizations, i.e., the differences from the base applications, distributed in the form of a template to another user having the same base applications on his/her system. A template builder utility allows a user to select a base application, customize the application, and store the customizations as a template. The template builder utility provides a graphical user interface (GUI), which enables users to perform a variety of different actions (i.e. New Template, Open Template, Save, Create Distribution Pack, and Exit Template Builder). After the user has specified the customizations within the application using the GUI and an editor and saved the customizations as a template using the template builder utility, the template is stored in a template storage file with the assistance of a template storage dynamic link library. Buxton does not describe or suggest receiving output from a command line utility but rather teaches that the user through a GUI and editor performs customizations to the base application. Column 13, lines 8-14 describes creation of templates using dynamic link libraries (DLL) that include formatting and storage retrieval functions and procedures to manage and simplify the storage and registration of templated applications for the template builder, template loader and template installer. Thus, column 13, lines 8-14 or any other part of the Buxton reference does not describe or suggest receiving output from a command line utility or storing the output in a system storage at a location identified by the identifier.

Claims 2-14, 16-20 and 22 depend from independent claims 1, 15 and 21, respectively. Accordingly, Applicant requests reconsideration and withdrawal of the rejection for claims 2-14, 16-20 and 22 for the reasons discussed above with respect to claims 1, 15 and 21.

All independent claims are allowable for the reasons set forth above. Dependent claims are allowable for at least the same reasons as corresponding independent claims.

Allowance of all claims is respectfully requested. The Commissioner is authorized to charge any additional fees and/or credit any overpayment to Deposit Account 20-1504 (MCT.0132US).

Respectfully submitted,

Date: 11/25/02

Dan C. Hu
Registration No. 40,025
TROP, PRUNER & HU, P.C.
8554 Katy Freeway, Suite 100
Houston, Texas 77024
(713) 468-8880 [Phone]
(713) 468-8883 [Fax]

21906
PATENT TRADEMARK OFFICE

APPENDIX

IN THE SPECIFICATION:

Please amend the paragraph starting at page 7, line 3, as follows:

--Referring now to FIG. 4, illustrative computer system 400 in accordance with one embodiment of the invention includes redirection routine [400] 402 (e.g., a routine in accordance with 200 and/or 300) to redirect output from a command line utility to a specified operating system controlled memory location. As shown, routine [400] 402 may be retained in storage device 404 which is coupled to processor 406 via system bus 408. It will be understood that storage device 404 may represent non-volatile memory devices or a combination of volatile and non-volatile memory devices. Illustrative non-volatile storage devices include, but not limited to: semiconductor memory devices such as EPROM, EEPROM, and flash devices; magnetic disks (fixed, floppy, and removable); other magnetic media such as tape; and optical media such as CD-ROM disks. It will be further recognized that computer system 400 may incorporate one or more input-output devices 410 such as one or more secondary bus bridge circuits, memory controllers, accelerated graphics port devices and network interface adapters.--

VERSION OF CLAIMS INDICATING CHANGES

Amend the following claims as indicated (unamended claims in smaller font):

1. A method to provide process command line utility output, comprising:
receiving an identifier;
receiving output from a command line utility; and
storing the command line utility output in a system storage at a location identified by the identifier.

2. The method of claim 1, wherein the act of receiving an identifier comprises receiving an identifier that identifies one or more entries in a system registry database.

3. The method of claim 2, wherein the act of receiving an identifier comprises receiving a root key identifier.

4. The method of claim 3, wherein the act of receiving a root key identifier further comprises receiving a sub-key identifier.

5. (Amended) The method of claim 2, wherein the system registry comprises an [a WINDOWS] operating system registry database.

6. The method of claim 1, wherein the act of receiving an identifier further comprises receiving a system storage identifier.

7. The method of claim 6, wherein the act of receiving a system storage identifier comprises receiving an identifier indicating a system registry.

8. The method of claim 6, wherein the act of receiving a system storage identifier comprises receiving an identifier indicating shared system memory.

9. The method of claim 8, wherein the act of receiving an identifier indicating shared system memory identifies a system clipboard memory.

10. The method of claim 1, wherein the act of receiving output from a command line utility comprises receiving output directly from the command line output utility.

11. The method of claim 1, wherein the act of receiving output from a command line utility comprises receiving output from the command line output utility through a subsequent command line output routine.

12. The method of claim 1, wherein the act of storing comprises associating each line of command line utility output with a line identifier in the system storage.

13. The method of claim 12, further comprising setting each line identifier to a value corresponding to that lines position in the command line utility output.

14. The method of claim 12, further comprising setting a default value of the received identifier to equal the total number of command utility output lines stored in the system storage.

15. A program storage device, readable by a computer, comprising instructions stored on the program storage device for causing the computer to:

receive an identifier;

receive output from a command line utility; and

store the command line utility output in system storage at a location identified by the identifier.

16. The program storage device of claim 15 wherein the instructions to store comprise instructions to store command line utility output in an operating system registry database.

17. The program storage device of claim 15 wherein the instructions to store comprise instructions to store command line utility output in an operating system maintained volatile memory.

18. The program storage device of claim 15 wherein the instructions to receive output comprise instructions to receive one or more lines of output from the command line utility, and the instructions to store further comprise instructions to store each of said one or more lines of output in the system storage.

19. The program storage device of claim 18 wherein the instructions to store further comprise instructions to associate a unique identifier with each of the one or more lines of output stored in the system storage.

20. The program storage device of claim 18 wherein the instructions to store further comprise instructions to set a value associated with the received identifier in the system storage equal to the number of lines of output stored in the system storage.

21. A computer system, comprising:

a processor; and

a storage device coupled to the processor, the storage device having stored thereon a program having instructions to receive an identifier, receive output from a command line utility, and store the command line utility output in system storage at a location identified by the identifier.

22. The computer system of claim 21, wherein the program comprises a dynamic link library.